KEYWORD EXTRACTION FROM THE NEWS FEED OR BLOG POSTS WITH IBM CLOUD

1.Introduction

a) Overview: Keyword extraction is a text analysis technique that automatically extracts the most used and most important words and expressions from a text. It helps summarize the content of texts and recognize the main topics discussed.

Keyword extraction uses machine learning artificial intelligence (AI) with natural language processing (NLP) to break down human language so that it can be understood and analysed by machines. It‟s used to find keywords from all manner of text: regular documents and business reports, social media comments, online forums and reviews, news reports, and more.

And In this project flask frames have been used to create an app and run the application from the browser itself. Like taking input from users like text from user and extracting minimum no of keywords from the text file and printing the keywords on to the keywords.html file.

Mainly in this project we are using rapid api which helps us to extract keywords and remove non keywords. When connecting an API to a project or application, you must have an API key to authenticate your request. Creating an app within RapidAPI generates an API key (X RapidAPI-Key) specific to that application. You can view analytics based on the API calls you make using this app key.

Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object
relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

b). Purpose: Purpose of the project is to extract keywords from blog posts as well as from any text file. When we want to understand key information from specific documents, we typically turn towards keyword extraction. Keyword extraction is the automated process of extracting the words and phrases that are most relevant to an input text.

It is a text analysis technique. We can obtain important insights into the topic within a short span of time. It helps concise the text and obtain relevant keywords. It saves the time of going through the entire document. Example use-cases are finding topics of interest from a news article and identifying the problems based on customer reviews and so. One of the techniques used for Keyword Extraction is TF-IDF (Term Frequency – Inverse Document Frequency)]

- Keywords themselves can be useful, particularly in formulating a response to “What

are people most frequently talking/asking about?”

· Keywords can help you focus in on smaller sets of individual records in order to learn more about them and begin to answer particular questions about user needs and goals · Keywords in combination with analysis of smaller sets of individual records can help you identify gaps in your understanding of users that can help focus subsequent research efforts

2. Literature Survey

In 1995, J.D. Cohen proposed an approach to draw index terms from text [4]. It doesn't use any stop list, stemmer, or any language and domain-specific component, allowing for easy application in any language or domain with slight modification. The method utilizes n-gram counts, which results in a function similar and more general than a stemmer. In 2002, M. Orton et al. demonstrated that important words of a text have a tendency to attract each other and form clusters [5]. He argues that the standard deviation of the distance between successive occurrences of a word is such a parameter to quantify this self-attraction.

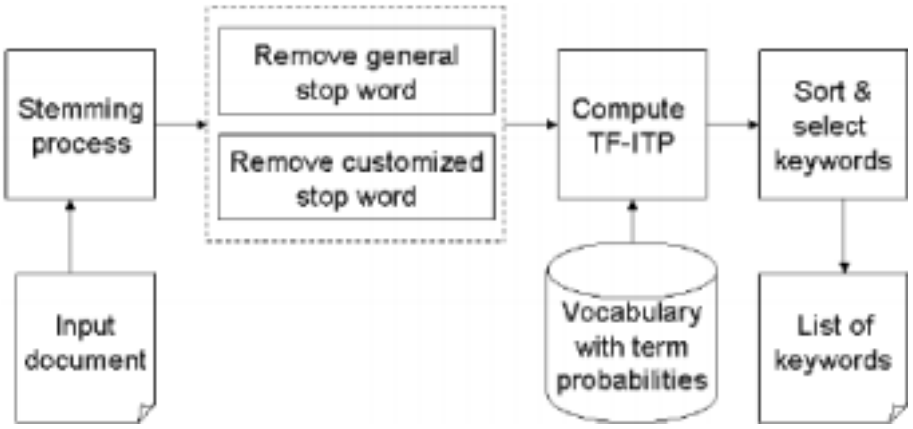
P. Carpena et al. proposed to automatically extract keywords from literary texts through a generalization of the level statistics analysis of quantum disordered systems [7]. They consider frequencies of the words along with their spatial distribution along the text, and are based on the observation that important words are significantly clustered whereas irrelevant words are distributed randomly in the text. No reference corpus is needed in this approach and it is especially suitable for single documents for which no priori information is available.

a)Existed System: The existed system is not able work on all type of data like blog posts must considering only text files whereas coming to this project it is about to extract keywords from blog posts also and also not been deployed with IBM cloud and also not used any frameworks like flask ,rapid api. Not able to extract keywords accurately.

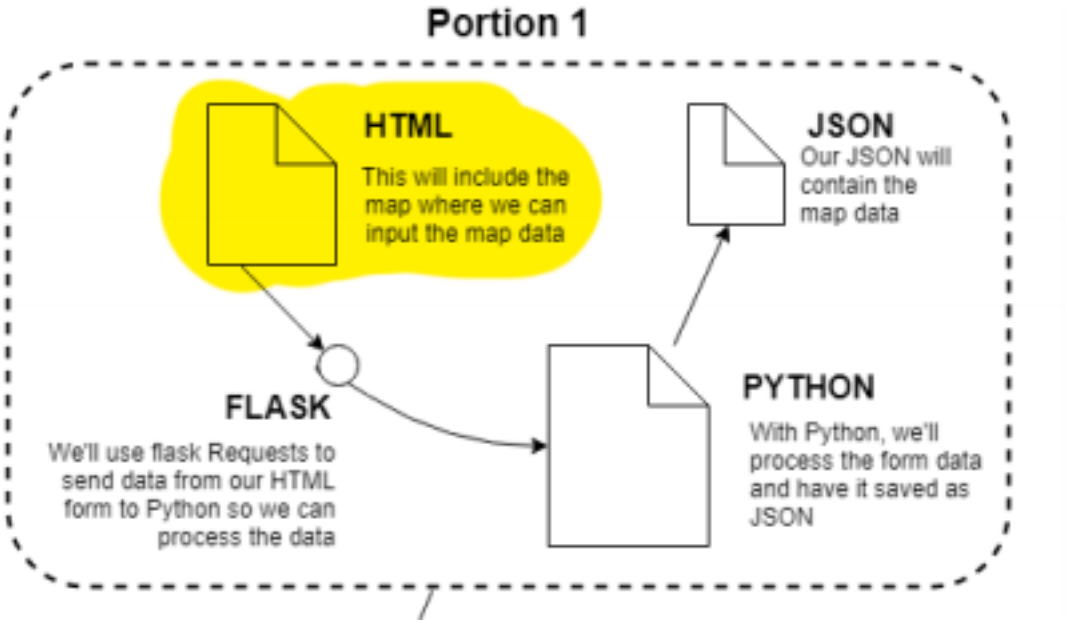
b) Proposed System: Keyword extraction is the automated process of extracting the most relevant words and expressions from the text. With more than 290 billion emails sent and received on a daily basis, and half a million tweets posted every single minute, using machines to analyse huge sets of data and extract important information is definitely a game-changer. We are building a flask application by using a keywords extraction API. This API uses advanced Natural Language Processing techniques to extract the most important 'Keywords' and 'Phrases' from a LIST of text or URL that you provide. It can take any type of text as input and will perform best on URLs corresponding to news, blog, content, etc. It can also take MULTIPLE text or URLs as input and you can specify the number of keywords that need to be extracted.

3. Theoretical Analysis:

a)Block Diagram:



b) **Hardware / Software designing**



4. Experimental Investigations:

Experimental Investigations refers to research papers which helped us to implement with the help of flask framework as well as usage of rapid API. And also using IBM cloud which is a good platform which helps to deploy models.

IBM Watson on the IBM Cloud helps to transform businesses, enhancing competitive advantage and disrupting industries by unlocking the potential within unstructured data. Fundamental to providing a strong foundation for companies wanting to leverage Watson IBM uses best-in-class security and compliance processes that allow for successful execution of challenging workloads.

IBM Cloud Deployment Services (ICDS) reduces complexity by integrating multi cloud environments with a single orchestration platform. That easily integrates with existing tools and architectures and is technology agnostic.

ICDS improves DevOps through automated patterns and workflows, boosts operational

efficiency and reduces services deployment time. Zero touch IT helps with resource management and tracks governance issues with a self-service portal. Get virtualization and container orchestration with cloud native infrastructure support by building your own platforms.

5. Flowchart

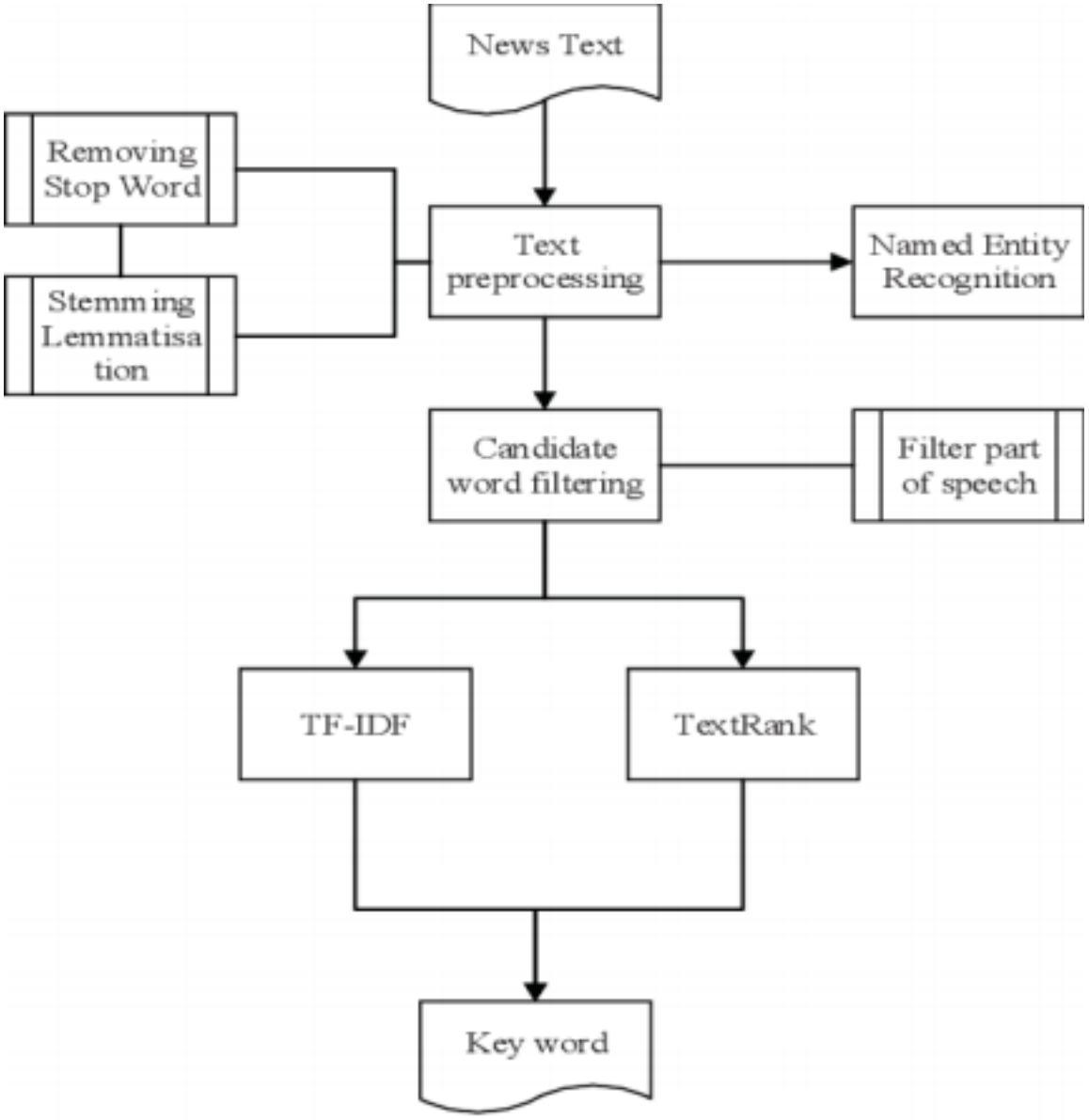


Fig. 2. Keyword Extraction Flow Chart

6.Result:

Gives output a set of keywords on the keywords.html file of required count of keywords .And also flowchart from IBM cloud platform on deploying the model. UI pages with a nice layout with different buttons which helps to take input from a text file and extract the data and print in the json format.

7. Advantages & Disadvantages:

Advantages: Advantages refers to deploying the model and also no need of more python code and also using API can easily find out keywords. Helps to find the keywords to find the recommended systems like finding the most watched or most buying products so helps for suggestions in the recommended systems. Keywords mainly help to find a video or a product based on keywords .

Disadvantages: Disadvantages are less in number as deploying the model with IBM cloud helps to reduce error in finding the keywords which or not. Suggesting fake videos or fake products or fake movies etc. Mainly in Recommendation systems.

8. Applications:

1. Movie Recommendation System

2. Sentiment Analysis

3.Youtube video statistics.

4. finding required documents

5. Groceries recommendation system

9. Conclusion

By the end of the project we had implemented the project successfully without any errors with a beautiful UI and developed an app for taking text as input and find the keywords. Mainly this project consists of home.html, extractor.html, keywords.html and extractor.py These are the files which helped to find the keywords and deployed with IBM cloud.

10. Future Scope

This can be implemented in more detail like adding a url which will redirect to that blog and extract the text and find the keywords and place it in the json file and print them on keywords.html file. Using logic to find the keywords rather than using the IBM WATSON API.

11. Appendix

a) Source code

Extraction.py

```
from flask import Flask, request, render_template

import re

import requests

import json

from ibm_watson import NaturalLanguageUnderstandingV1

from ibm_cloud_sdk_core.authenticators import IAMAuthenticator

from ibm_watson.natural_language_understanding_v1 import Features,
KeywordsOptions



# watson api for Keyword Extraction

authenticator =
IAMAuthenticator('OxgvILpwXSEyaOUNrMMg7aQQeeB8lfoEqg-dJ-_T-boj')

natural_language_understanding = NaturalLanguageUnderstandingV1(

   version='2021-03-25',

   authenticator=authenticator

)



natural_language_understanding.set_service_url('https://api.eu-gb.natural-lan
guage-understanding.watson.cloud.ibm.com/instances/e29dd273-2c9a-4e3c-882f-7b
98fe7053ea')







app = Flask(__name__)



@app.route('/')

def home():

   return render_template('home.html')
```

```
@app.route('/extractor')

def extractor():

   return render_template('extractor.html')



@app.route('/keywords', methods=['POST'])

def keywords():

   sen = request.form['output']

   type = request.form['type']

   # num = request.form['wordnum']

   keyword = check(sen,type)

   return render_template('keyword.html', keyword=keyword)



def check(sentence,type):

   # url =
"https://textanalysis-keyword-extraction-v1.p.rapidapi.com/keyword-extractor-
text"

   # payload = "text=" + sentence + "&wordnum=" + str(500)

   # headers = {

   #     'content-type': "application/x-www-form-urlencoded",

   #     'x-rapidapi-key':
"bdf00f69d0mshf71b715e665de84p187496jsn103deb88c72c",

   #     'x-rapidapi-host':
"textanalysis-keyword-extraction-v1.p.rapidapi.com"

   #     }

   #

   # response = requests.request("POST", url, data=payload, headers=headers)

   # print(response.text)



   if(type == "url"):
```

```
        response = natural_language_understanding.analyze(

            # url='www.ibm.com',

            url=sentence,

            features=Features(keywords=KeywordsOptions(sentiment=False,
emotion=False))).get_result()

    else :

        response = natural_language_understanding.analyze(

            # url='www.ibm.com',

            text=sentence,

            features=Features(keywords=KeywordsOptions(sentiment=False,
emotion=False))).get_result()

    k = []

    for i in response['keywords']:

        k.append(i['text'])

    return k


if __name__ == "__main__":

    app.run(debug="True")
```

home.html

```
<!DOCTYPE html>


<html >
```

```
<head>

 <meta charset="UTF-8">

 <meta name="viewport" content="width=device-width, initial-scale=1">

 <title> Keyword Extraction from news feed or blogposts</title>

 <link href='https://fonts.googleapis.com/css?family=Pacifico'
rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet'
type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Hind:300'
rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300'
rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Merriweather'
rel='stylesheet'>

<link href='https://fonts.googleapis.com/css?family=Josefin Sans'
rel='stylesheet'>

<link href='https://fonts.googleapis.com/css?family=Montserrat'
rel='stylesheet'>
```

```
<style>

.header {

        top:0;

        margin:0px;

        left: 0px;

        right: 0px;

        position: fixed;

        background-color: #091425;

        color: white;

        box-shadow: 0px 8px 4px grey;

        overflow: hidden;

        padding-left:20px;

        font-family: 'Josefin Sans';
```

```
        font-size: 2vw;

        width: 100%;

        height:8%;

        text-align: center;

    }

    .topnav {

 overflow: hidden;

 background-color: #333;

}



.topnav-right a {

 float: left;

 color: #f2f2f2;
```

```
 text-align: center;

 padding: 14px 16px;

 text-decoration: none;

 font-size: 18px;

}

.topnav-right a:hover {

 background-color: #ddd;

 color: black;

}

.topnav-right a.active {

 background-color: #565961;
```

```
 color: white;

}

.topnav-right {

 float: right;

 padding-right:100px;

}

body {

 background-color:#ffffff;

 background-repeat: no-repeat;

 background-size:cover;
```

```
background-position: 0px 0px;

}

.button {

background-color: #091425;

border: none;

color: white;

padding: 15px 32px;

text-align: center;

text-decoration: none;

display: inline-block;

font-size: 16px;

border-radius: 12px;

}
```

```
.button:hover {

 box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);

}

form {border: 3px solid #f1f1f1; margin-left:400px;margin-right:400px;}



input[type=text], input[type=password] {

 width: 100%;

 padding: 12px 20px;

 display: inline-block;

 margin-bottom:18px;

 border: 1px solid #ccc;

 box-sizing: border-box;

}
```

```
button {

 background-color: #091425;

 color: white;

 padding: 14px 20px;

 margin-bottom:8px;

 border: none;

 cursor: pointer;

 width: 17%;

 border-radius:4px;

 font-family:Montserrat;

}
```

```
button:hover {

 opacity: 0.8;

}



.cancelbtn {

 width: auto;

 padding: 10px 18px;

 background-color: #f44336;

}



.imgcontainer {

 text-align: center;

 margin: 24px 0 12px 0;
```

```
}

img.avatar {
 width: 30%;
 border-radius: 50%;
}

.container {
 padding: 16px;
}

span.psw {
 float: right;
```

```
 padding-top: 16px;

}



/* Change styles for span and cancel button on extra small screens */

@media screen and (max-width: 300px) {

 span.psw {

    display: block;

    float: none;

 }

 .cancelbtn {

    width: 100%;

 }

}
```

```
.home{

  margin:80px;




 width: 84%;

 height: 500px;

 padding-top:10px;

 padding-left: 30px;




}

.login{

  margin:80px;

  box-sizing: content-box;
```

```
 width: 84%;

 height: 420px;

 padding: 30px;

 border: 10px solid blue;

}

.left,.right{

box-sizing: content-box;

height: 400px;

margin:20px;

border: 10px solid blue;

}



.mySlides {display: none;}
```

```
img {vertical-align: middle;}



/* Slideshow container */

.slideshow-container {

 max-width: 1000px;

 position: relative;

 margin: auto;

}



/* Caption text */

.text {

 color: #f2f2f2;

 font-size: 15px;
```

```
 padding: 8px 12px;

 position: absolute;

 bottom: 8px;

 width: 100%;

 text-align: center;

}

/* The dots/bullets/indicators */

.dot {

 height: 15px;

 width: 15px;

 margin: 0 2px;

 background-color: #bbb;

 border-radius: 50%;
```

```
 display: inline-block;

 transition: background-color 0.6s ease;

}

.active {

 background-color: #717171;

}

/* Fading animation */

.fade {

 -webkit-animation-name: fade;

 -webkit-animation-duration: 1.5s;

 animation-name: fade;
```

```
 animation-duration: 1.5s;

}



@-webkit-keyframes fade {

 from {opacity: .4}

 to {opacity: 1}

}



@keyframes fade {

 from {opacity: .4}

 to {opacity: 1}

}
```

```
/* On smaller screens, decrease text size */

@media only screen and (max-width: 300px) {

 .text {font-size: 11px}

}

</style>

</head>

<body style="font-family:'Times New Roman', Times,
serif;background-color:#ffffff;">

<div class="header">

<div style="width:50%;float:left;font-size:2vw;text-align:left;color:white;
padding-top:1%;padding-left:5%;">Keyword Extraction from news feed or
blogposts</div>

 <div class="topnav-right"style="padding-top:0.5%;">
```

```
   <a class="active" href="home.html">Home</a>


   <a href="/extractor" >Extractor</a>


 </div>


</div>




<div style="background-color:#ffffff;">


<div style="width:60%;float:left;">


<div
style="font-size:40px;font-family:Montserrat;padding-left:20px;text-align:cen
ter;padding-top:10%;">


<b>Analyze thousands of lines in seconds!!</b></div><br><br><br>


<div
style="font-size:20px;font-family:Montserrat;padding-left:70px;padding-right:
30px;text-align:justify;"> Imagine you want to analyze thousands of online
reviews about your


blog. Keyword extraction helps you sift through the whole set of data
```

and obtain the words that best describe each review in just seconds.

That way, you can easily and automatically see what your viewers are

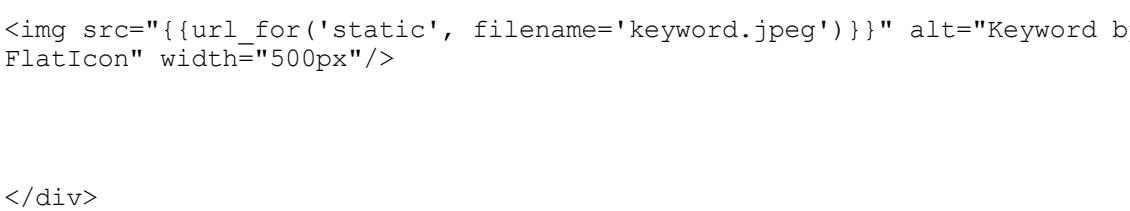
mentioning most often, saving your teams hours upon hours of

manual processing. This application saves lots of your time by quick

analysis of text.

<div
style="font-size:20px;font-family:Montserrat;padding-left:70px;padding-right:
30px;">Click here to

<button type="submit">Extract</button>



```
</div>
```

```
<div class="home">
```

```
<br>
```

```
</div>
```

```
</body>
```

```
</html>
```

Extraction.html

```
<!DOCTYPE html>
```

```
<html >
```

```
<head>
```

```
 <meta charset="UTF-8">

 <meta name="viewport" content="width=device-width, initial-scale=1">

 <title> Keyword Extraction from news feed or blogposts</title>

 <link href='https://fonts.googleapis.com/css?family=Pacifico'
rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet'
type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Hind:300'
rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300'
rel='stylesheet' type='text/css'>

<link href='https://fonts.googleapis.com/css?family=Merriweather'
rel='stylesheet'>

<link href='https://fonts.googleapis.com/css?family=Josefin Sans'
rel='stylesheet'>

<link href='https://fonts.googleapis.com/css?family=Montserrat'
rel='stylesheet'>

<style>
```

```
.header {

        top:0;

        margin:0px;

        left: 0px;

        right: 0px;

        position: fixed;

        background-color: #091425;

        color: white;

        box-shadow: 0px 8px 4px grey;

        overflow: hidden;

        padding-left:20px;

        font-family: 'Josefin Sans';

        font-size: 2vw;
```

```
        width: 100%;

        height:8%;

        text-align: center;

    }

    .topnav {

 overflow: hidden;

 background-color: #333;

}



.topnav-right a {

 float: left;

 color: #f2f2f2;

 text-align: center;
```

```
 padding: 14px 16px;

 text-decoration: none;

 font-size: 18px;

}

.topnav-right a:hover {

 background-color: #ddd;

 color: black;

}

.topnav-right a.active {

 background-color: #565961;

 color: white;
```

```
}



.topnav-right {

 float: right;

 padding-right:100px;

}



body {



 background-color:#ffffff;

 background-repeat: no-repeat;

 background-size:cover;

 background-position: 0px 0px;
```

```
}

.button {

background-color: #091425;

border: none;

color: white;

padding: 15px 32px;

text-align: center;

text-decoration: none;

display: inline-block;

font-size: 16px;

border-radius: 12px;

}

.button:hover {
```

```
 box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);

}

form { margin-left:400px;margin-right:400px;}



select {

  width: 50%;

  margin-bottom: 10px;

  background: rgba(255,255,255,255);

  outline: none;

  padding: 10px;

  font-size:1.2vw;

  color: #000000;

  text-shadow: 1px 1px 1px rgba(0,0,0,0.3);
```

```
  border:3px solid #ddd;

  border-radius:4px;

  font-family:Montserrat;

  margin-left: 9%;

}



input[type=text]{

 width: 100%;

 padding: 12px 20px;

 display: inline-block;

 margin-bottom:18px;

 border: 3px solid #ddd;

 border-radius:4px;
```

```
  font-family:Montserrat;

 box-sizing: border-box;

}



textarea {

 width: 100%;

 padding: 12px 20px;

 display: inline-block;

 margin-bottom:18px;

 border: 3px solid #ddd;

 border-radius:4px;

 box-sizing: border-box;

 font-family:Montserrat;
```

```
}

button {
 background-color: #091425;
 color: white;
 padding: 14px 20px;
 margin-bottom:8px;
 border: none;
 cursor: pointer;
 width: 15%;
 border-radius:4px;
 font-family:Montserrat;
}
```

```
button:hover {

 opacity: 0.8;

}



.cancelbtn {

 width: auto;

 padding: 10px 18px;

 background-color: #f44336;

}



.imgcontainer {

 text-align: center;
```

```
 margin: 24px 0 12px 0;

}



img.avatar {

 width: 30%;

 border-radius: 50%;

}



.container {

 padding: 16px;

}



span.psw {
```

```
 float: right;

 padding-top: 16px;

}

/* Change styles for span and cancel button on extra small screens */

@media screen and (max-width: 300px) {

 span.psw {

    display: block;

    float: none;

 }

 .cancelbtn {

    width: 100%;

 }
```

```
}



.home{

  margin:80px;



 width: 84%;

 height: 500px;

 padding-top:10px;

 padding-left: 30px;



}

ft,.right{

box-sizing: content-box;
```

```
height: 400px;

margin:20px;

border: 10px solid blue;

}

/* The dots/bullets/indicators */

.dot {

 height: 15px;

 width: 15px;

 margin: 0 2px;

 background-color: #bbb;

 border-radius: 50%;
```

```
 display: inline-block;

 transition: background-color 0.6s ease;

}

.active {

 background-color: #717171;

}

/* On smaller screens, decrease text size */

@media only screen and (max-width: 300px) {

 .text {font-size: 11px}

}
```

```
</style>


</head>




<body style="font-family:'Times New Roman', Times,
serif;background-color:#ffffff;overflow: scroll;">




<div class="header">


<div style="width:50%;float:left;font-size:2vw;text-align:left;color:white;
padding-top:1%;padding-left:8%;">Keyword Extraction from news feed or
blogposts</div>


 <div class="topnav-right"style="padding-top:0.5%;">




   <a class="active" href="/">Home</a>


 </div>


</div>
```

```
<div style="background-color:#ffffff;">


<div style="width:100%;float:left;">







<div
style="font-size:27px;font-family:Montserrat;padding-left:10%;text-align:left
;padding-top:1%;">


<b>Give the newsfeed/blog you want to extract keywords from</b></div><br><br>


<form action="/keywords" method="post"
style="font-size:20px;font-family:Montserrat;margin-left:0%;margin-right:0%;p
adding:0%;text-align:justify;">


<select name="type" id="type">







                <option value="select" selected>Select an option</option>


                 <option value="url">Would you like to summarize article from
a website?</option>
```

```
                <option value="text">Would you like to specify the text on
your own?</option>









</select>



<br><br>



<div id="output" name="output"
style="font-size:20px;font-family:Montserrat;margin-left:8%;margin-right:10%;
padding:1%;text-align:justify;">



<br>

















</div>









<div
style="font-size:20px;font-family:Montserrat;padding-left:9%;padding-right:30
px;">
```

```
<button type="submit">Extract</button>

</div>

<br><br>

</div>

</div>

<div class="home">

<br>
```

```
</div>

</body>

<script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></scr
ipt>

<script>

$(document).ready(function () {

   $("#type").change(function () {

       var val = $(this).val();

       if (val == "url") {

           $("#output").html("<input type='text' name='output'
placeholder='Specify the URL'>");
```

```
        } else if (val == "text") {


            $("#output").html("<textarea id='txtArea' rows='12' cols='163'
name='output' placeholder='Give your article here . .'
autofocus></textarea>");


        }


    });


});


</script>


</html>
```

Keywords.html

```
<!DOCTYPE html>

<html >


<head>

 <meta charset="UTF-8">

 <meta name="viewport" content="width=device-width,

initial-scale=1">

 <title> Keyword Extraction from news feed or
```

```
blogposts</title>
 <link
href='https://fonts.googleapis.com/css?family=Pacifico'
rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo'
rel='stylesheet' type='text/css'>
<link
href='https://fonts.googleapis.com/css?family=Hind:300'
rel='stylesheet' type='text/css'>
<link
href='https://fonts.googleapis.com/css?family=Open+Sans+Cond
ensed:300' rel='stylesheet' type='text/css'>
<link
href='https://fonts.googleapis.com/css?family=Merriweather'
rel='stylesheet'>
<link href='https://fonts.googleapis.com/css?family=Josefin
Sans' rel='stylesheet'>
<link
href='https://fonts.googleapis.com/css?family=Montserrat'
rel='stylesheet'>
<style>
.header {
        top:0;
        margin:0px;
        left: 0px;
        right: 0px;
        position: fixed;
```

```
        background-color: #091425;
        color: white;
        box-shadow: 0px 8px 4px grey;
        overflow: hidden;
        padding-left:20px;
        font-family: 'Josefin Sans';
        font-size: 2vw;
        width: 100%;
        height:8%;
        text-align: center;
    }
    .topnav {
 overflow: hidden;
 background-color: #333;
}

.topnav-right a {
 float: left;
 color: #f2f2f2;
 text-align: center;
 padding: 14px 16px;
 text-decoration: none;
 font-size: 18px;
}

.topnav-right a:hover {
```

```
 background-color: #ddd;
 color: black;
}

.topnav-right a.active {
 background-color: #565961;
 color: white;
}

.topnav-right {
 float: right;
 padding-right:100px;
}

body {

 background-color:#ffffff;
 background-repeat: no-repeat;
 background-size:cover;
 background-position: 0px 0px;
 }
 .button {
 background-color: #091425;
 border: none;
 color: white;
 padding: 15px 32px;
```

```
 text-align: center;
 text-decoration: none;
 display: inline-block;
 font-size: 16px;
 border-radius: 12px;
}
.button:hover {
 box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0
rgba(0,0,0,0.19);
}
form {border: 3px solid #f1f1f1;
margin-left:400px;margin-right:400px;}

input[type=text], input[type=password] {
 width: 100%;
 padding: 12px 20px;
 display: inline-block;
 margin-bottom:18px;
 border: 1px solid #ccc;
 box-sizing: border-box;
}

button {
 background-color: #091425;
 color: white;
 padding: 14px 20px;
```

```
 margin-bottom:8px;
 border: none;
 cursor: pointer;
 width: 15%;
 border-radius:4px;
 font-family:Montserrat;
}

button:hover {
 opacity: 0.8;
}

.cancelbtn {
 width: auto;
 padding: 10px 18px;
 background-color: #f44336;
}

.imgcontainer {
 text-align: center;
 margin: 24px 0 12px 0;
}

img.avatar {
 width: 30%;
 border-radius: 50%;
```

```
}

.container {
 padding: 16px;
}

span.psw {
 float: right;
 padding-top: 16px;
}

/* Change styles for span and cancel button on extra small
screens */
@media screen and (max-width: 300px) {
 span.psw {
    display: block;
    float: none;
 }
 .cancelbtn {
    width: 100%;
 }
}

.home{
  margin:80px;
```

```
 width: 84%;
 height: 500px;
 padding-top:10px;
 padding-left: 30px;

}
.login{
  margin:80px;
  box-sizing: content-box;
 width: 84%;
 height: 420px;
 padding: 30px;
 border: 10px solid blue;
}
.left,.right{
box-sizing: content-box;
height: 400px;
margin:20px;
border: 10px solid blue;
}

.mySlides {display: none;}
img {vertical-align: middle;}

/* Slideshow container */
.slideshow-container {
```

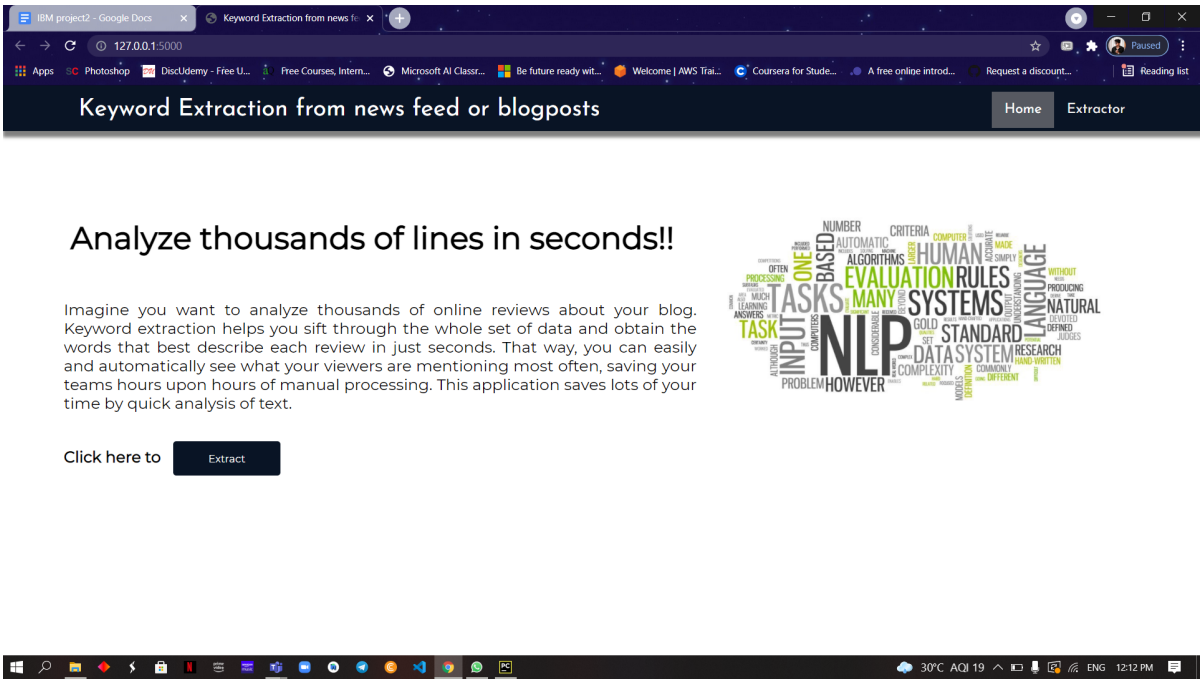
```
 max-width: 1000px;
 position: relative;
 margin: auto;
}

/* Caption text */
.text {
 color: #f2f2f2;
 font-size: 15px;
 padding: 8px 12px;
 position: absolute;
 bottom: 8px;
 width: 100%;
 text-align: center;
}
/* The dots/bullets/indicators */
.dot {
 height: 15px;
 width: 15px;
 margin: 0 2px;
 background-color: #bbb;
 border-radius: 50%;
 display: inline-block;
 transition: background-color 0.6s ease;
}
```

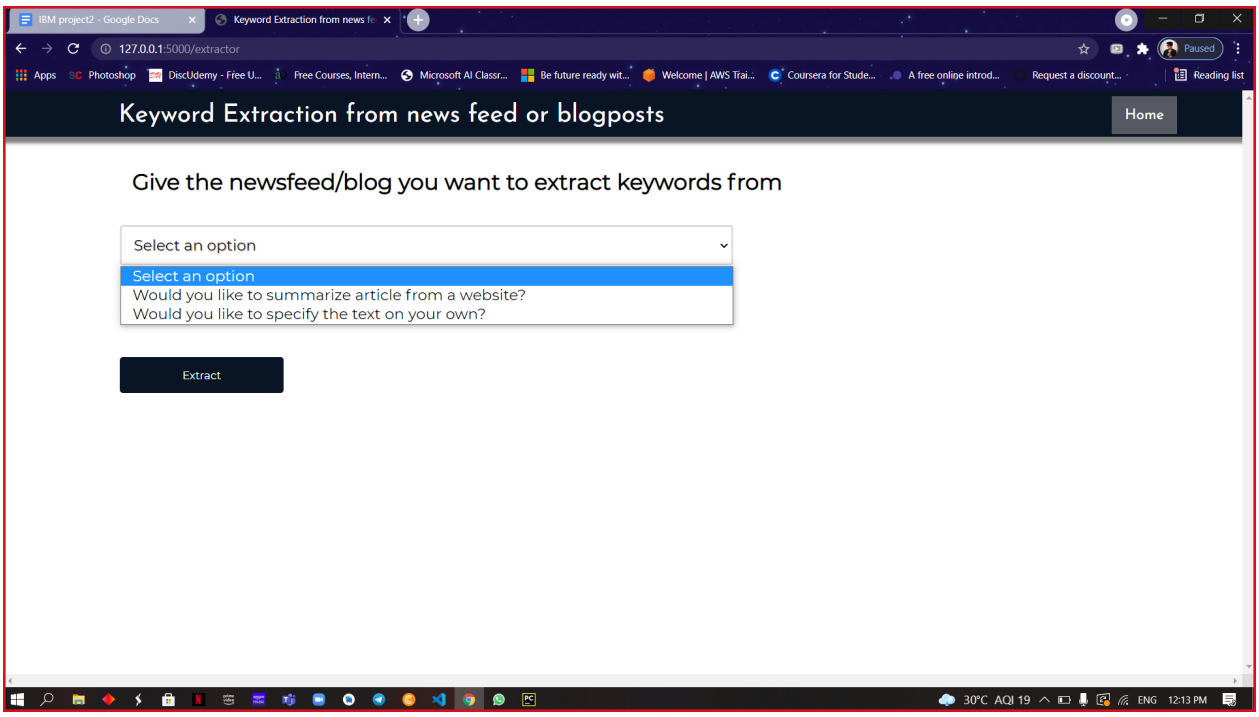
```
.active {
 background-color: #717171;
}

/* Fading animation */
.fade {
 -webkit-animation-name: fade;
 -webkit-animation-duration: 1.5s;
 animation-name: fade;
 animation-duration: 1.5s;
}

@-webkit-keyframes fade {
 from {opacity: .4}
 to {opacity: 1}
}

@keyframes fade {
 from {opacity: .4}
 to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
 .text {font-size: 11px}
}
```

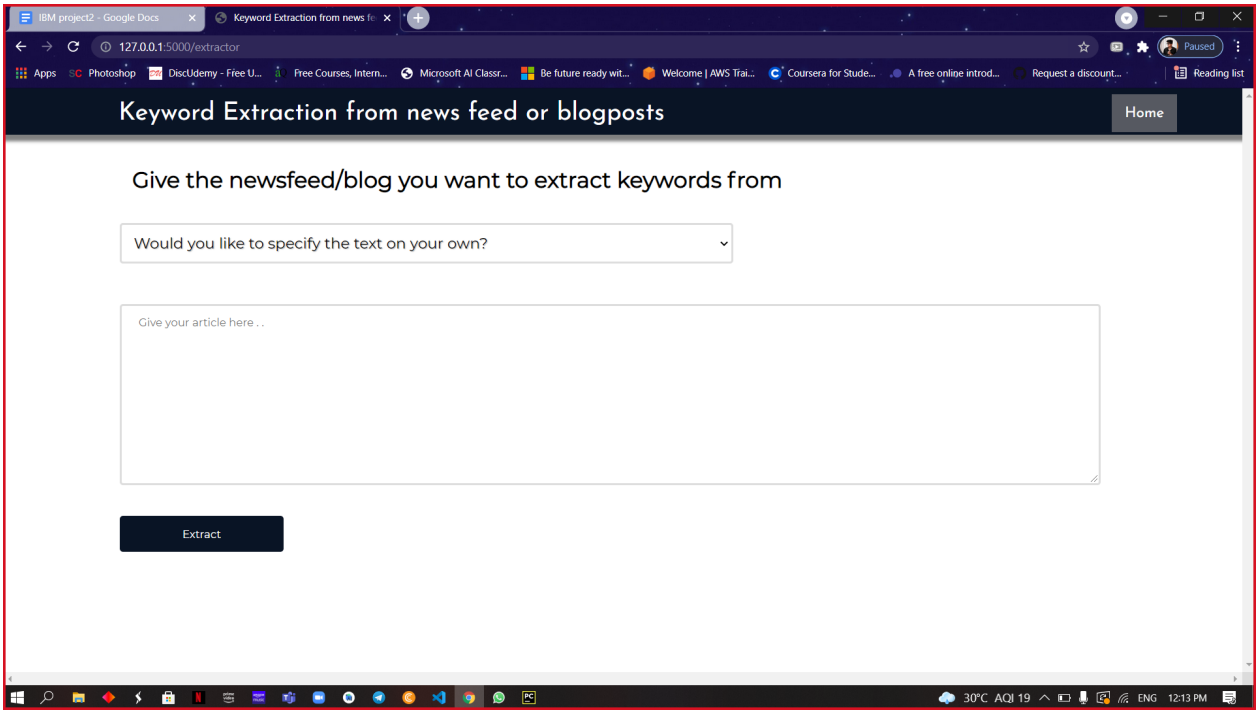
```
th, td {
 border-bottom: 1px solid #ddd;
}
</style>
</head>

<body style="font-family:'Times New Roman', Times,
serif;background-color:#ffffff;overflow: scroll;">

<div class="header">
<div
style="width:50%;float:left;font-size:2vw;text-align:left;co
lor:white; padding-top:1%;padding-left:5%;">Keyword
Extraction from news feed or blogposts</div>
 <div class="topnav-right"style="padding-top:0.5%;">

   <a  href="/">Home</a>
  <a href="/extractor">Extractor</a>
 </div>
</div>

<div style="background-color:#ffffff;">
<div style="width:100%;float:left;">
<div
style="font-size:27px;font-family:Montserrat;padding-left:10
%;text-align:left;padding-top:1%;">
<b>Keywords Extracted</b></div><br><br>
```

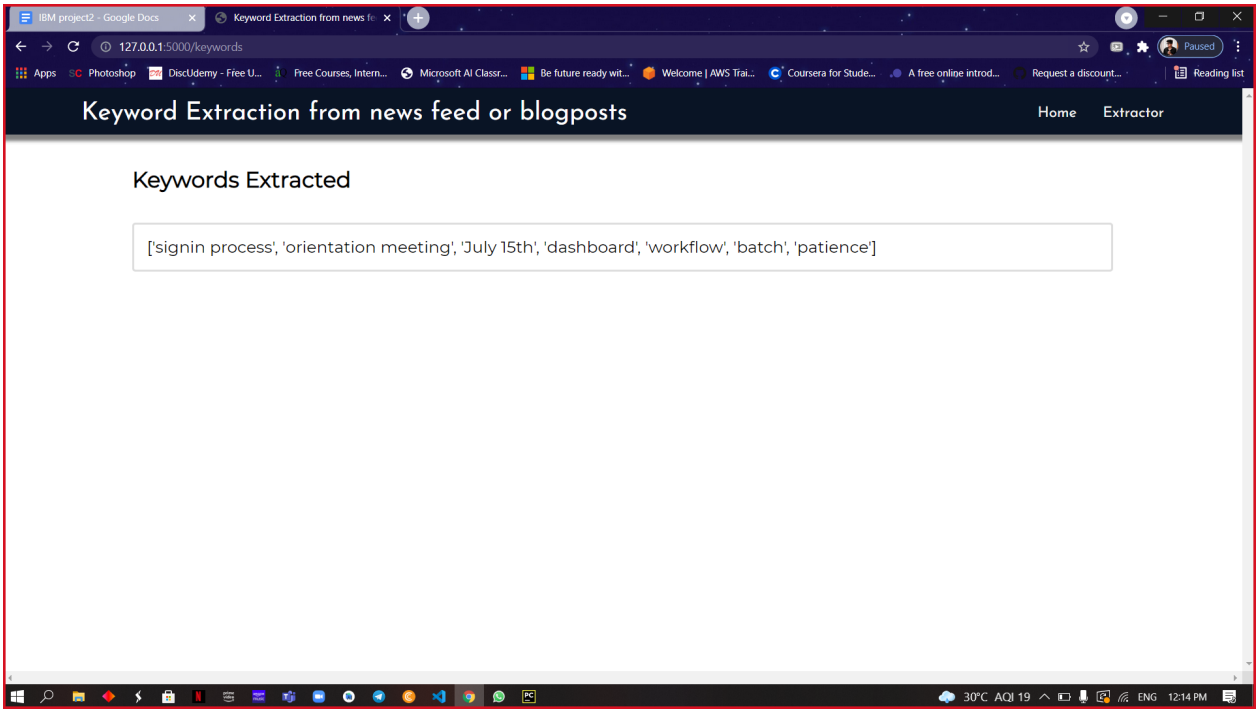
```
<div
style="font-size:20px;font-family:Montserrat;margin-left:10%
;margin-right:10%;padding:1%;text-align:justify;border:3px
solid #ddd;border-radius:4px;">
{{keyword}}
</div><br><br>



<br><br>


</div>


</div>


<div class="home">


<br>









</div>


</body>
</html>
```

b) UI output Screenshot.






IBM project2 - Google Docs
Keyword Extraction from news fe
127.0.0.1:5000/extractor
Paused
Apps
Photoshop
DiscUdemy - Free U...
Free Courses, Intern...
Microsoft AI Classr...
Be future ready wit...
Welcome | AWS Trai...
Coursera for Stude...
A free online introd...
Request a discount...
Reading list
Keyword Extraction from news feed or blogposts
Home
Give the newsfeed/blog you want to extract keywords from
Would you like to specify the text on your own?
Give your article here . .
Extract
30°C AQI 19
ENG
12:13 PM


IBM project2 - Google Docs
Keyword Extraction from news fe
127.0.0.1:5000/keywords
Paused
Apps
Photoshop
DiscUdemy - Free U...
Free Courses, Intern...
Microsoft AI Classr...
Be future ready wit...
Welcome | AWS Trai...
Coursera for Stude...
A free online introd...
Request a discount...
Reading list
Keyword Extraction from news feed or blogposts
Home
Extractor
Keywords Extracted
['signin process', 'orientation meeting', 'July 15th', 'dashboard', 'workflow', 'batch', 'patience']
30°C AQI 19
ENG
12:14 PM